SLUG NEWSLETTER

sinclair LOUISVILLE USERS GROUP

NOV 87

The Sinclair Louisville User Group meeting came to order on Sunday, October 11, 1987 at 2:00 PM at the Government Center on Outer Loop road.

If you have articles or programs that you wish to be included in the newsletter, please give them to Shawn Merrick. He will then edit them and pass them on to our publisher, Jim Dodrill. This is the way to submit articles or programs for publication in SLUG. Don't be a lurker, get your name in print!

Speaking of the newsletter, you can't help but notice the multicolored pages. Jim Dodrill thought it would "brighten" things up a bit, so there they are. By the way, Jim says he IS NOT colorblind, he just wanted to be as colorful as the trees are (or were). We need to set a meeting for his LARKEN disc system demo. He should be well familiar with its operation by now.

The ADAM made its debut' at our last meeting. Shawn Merrick brought his system in and treated us to a demo of ADAMCALC, a spreadsheet pgm, and a database program that is similar to PRO/FILE on the 2068. I can't remember the name of the database program, but it does a nice job of sorting and listing information. Also demoed was a game package. Shawn gave us some news of the BBS program he has written, and selling, for the ADAM machine. From the messages I've seen on Compuserve about Shawns BBS, it will be the super fast program with xmodem transfer that the ADAMites have been waiting for. Nice stuff, Shawn! Now for the phone line.....

For those of you who are planning to attend the T/S WINTERFEST in Florida, there is information on entrance fees and pre-registration inside this issue of newsletter. Dates are March 4-6, 1988, so make your plans now, and give yourself an after Christmas gift in March.

After Shawns demo, the meeting was officially ajourned at 3:45 PM. Start a New Years resolution early and come to the meeting next month on Sunday, December 13, 1987. It will be the last chance to see you this year.

Bob Crecco, Secretary

President	JIM PITTS	Editor	SHAWN MERRICK
Secretary	BOB CRECCO	Librarian	ED RADTKE
Treasurer	JOHN HAMILTION	Public Realtions	RAY MANLEY

THE LATEST INFORMATION FROM THE SUNSTATE BBS .....

C-T/S WINTERFEST NEWS

SUNSTATE Timex/Sinclair WINTERFEST '88 will be held on March 4-6, 1988 in sunny Orlando Fl. The Orlando Marriott Hotel will be the host hotel. Advanced reservations are now being accepted, both for entry to the WINTERFEST and vendor's booths. Prepaid entrance will be $5 single, $9 family, at the door $8 single, $12 family. SEE Message #2 for details on how to register.

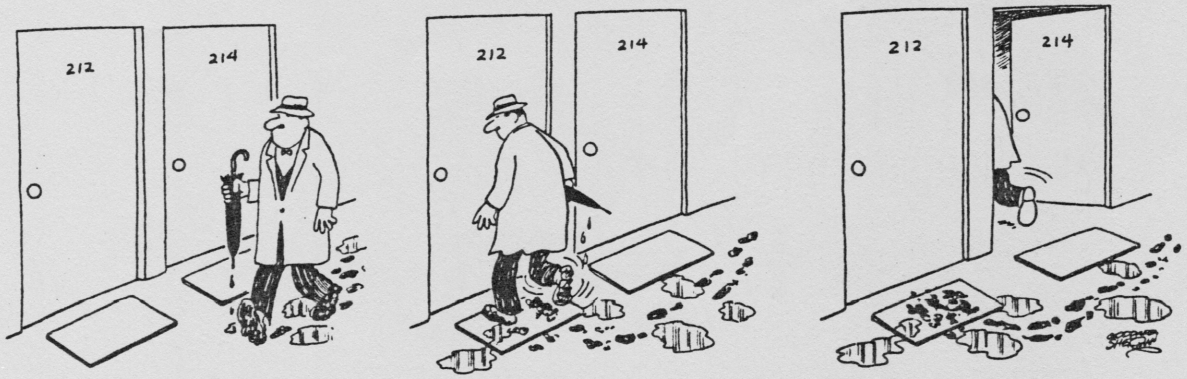
Message # 0002
September 1,1987
To: ALL
From: Old Man Time
Re: Winterfest registration

To pre-register for the WINTER FEST: Mail your name(s), Address, phone #, proper fee ($5 single,$9 family), TO:

SUNSTATE T/S WINTERFEST
249 North Harden Ave.
Orange City, Fl 32763
Mke checks payable to:
Northeast Florida T/S Users
Group (NEF T/S Users GRoup)

WE are using their checking account as they are one of the four host groups.

ARLO AND JANIS® by Jimmy Johnson
LISTEN MY CHILDREN, AND YOU SHALL HEAR,
OF A MODERN-DAY HERO LIKE PAUL REVERE.
HE RUNS THROUGH THE OFFICE, NOT THROUGH THE TOWN...
SCREAMING, "STORE YOUR COPY, ERE THE SYSTEM GOES DOWN!"
JOHNSON
10/5
H. KOHANE
© 1987 by NEA, Inc.



Coleco ADAM Review #3 by Shawn Merrick, November 1987

CP/M 2.2 with Assembler from Coleco Industries and Digital Research Incorporated, $25

CP/M is an operating system for 8080 and Z80 computers. It allows different brands or models with these CPU's to become highly compatible with each other. For example, many public domain CP/M programs require little or no changes to run on ADAM even though they were written on other computers.

Unlike CP/M 3.0 (plus) for the Commodore 128, ADAM CP/M 2.2 includes utilities and programming aids to get you started. It has programs for drive status, file copy, text editing (ASCII), and an assembler to create machine language programs.

Unfortunately, the assembler (ASM.COM) is limited to 8080 instructions (which work on Z80's) and can not take advantage of special Z80-ONLY operations. Many programmers think Z80 mnemonics (text instructions) are easier to understand than the 8080 versions, and Z80 offers more functions. There is a lot you can do with ASM.COM, but you probably won't need it after you get a good public domain Z80 assembler.

CP/M allows ADAM to run some superior-quality communication programs. MEX (Modem EXecutive) is one of the best. It can be programmed with instruction files to dial a number, log onto a system, and send or receive information. You can be asleep while all this is happening! These terminal programs often include the XMODEM file transfer protocal which allows error-free exchange of programs and files. XMODEM programs under the ADAM operating system are rare.

An excellent CP/M feature is the RAMDisk. If you have a 64K RAM expander installed, you will have 55K for temporary file storage. It is lightning-fast since no mechanical parts are involved. ORPHANWARE in Akron, Ohio makes 256 and 512K expanders providing the best speed from the biggest tasks.

CP/M is designed for 80 column systems. The ADAM version only displays 30 columns on the screen at a time. You can scroll to the other 50 columns, use a screen chop program, or buy an 80 column card and monochrome monitor.

The BIOS (Basic Input/Output System) is the only part of CP/M tailored for ADAM. It tells CP/M how ADAM hardware works. While there are a few annoying bugs in the BIOS, patches and fixes have been made available by generous ADAM programmers. Datapack users will find some problems with the FORMAT program and CP/M datapacks are more likely to get tangled during read errors. CP/M seems to rely heavily on the directory (middle of tape) and is tediously slower than ADAM operating system. The disk drives work beautifully and

are a worthwhile investment if you plan to use CP/M frequently.

The documentation is extensive for beginners, but expert programmers may be confused or upset with errors and ommisions in technical details. Close to 300 pages are divided into various sections in the vinyl binder.

In spite of the flaws I've mentioned, CP/M offers a lot of potential to ADAM users. Any one of the fine public domain programs can make it worth the purchase price.

Next review- SmartFiler

ADAM Print & Run, written by Shawn Merrick, August 1987

Here's a program that demonstrates how the ADAM computer can do 2 things at one time. When you run this, it will start printing the alphabet, and list the program on your screen twice during the printout! A machine language call is made to the START PRINT routine. This tells the 6801 in the printer to print up to 16 characters, and the Z80 continues to work (listing the program on the screen)! It's not really useful in itself, but serves as a model for using ADAM to it's potential.

```
]
  5 REM  Shawn M. Merrick
  6 REM  Print & Run demonstration
  8 REM  Machine Language routine
 10 DATA    33,17,0,1,16,0,62,2,205,194,251,201
 20 FOR q = 0 TO 11
 30 READ p: POKE 3+q, p
 40 NEXT q
 45 REM  CR,LF, and some alphabet
 50 DATA     13,10,65,66,67,68,69,70,71,72,73,74,75,76,77,78
 60 FOR x = 0 TO 15
 70 READ p: POKE 17+x, p
 80 NEXT
 90 CALL 3
100 LIST
110 LIST

]
```

3-D WIRE FRAME GRAPHICS
By Tim Swenson

The program listed below allows a person to define a 3-D object, draw it on the screen, and rotate it in three different planes.

The heart of the program lies in how the object is represented. Two arrays hold the information for the object. An vertex array (list) holds all of the vertices. An edge array (list) holds the two endpoints of the edges. The edgel array refers back to the vertex list for the actual points. It only holds the number of the vertex.

Lines 100 to 1030 in the program holds the definition of a cube and some values for variables used by the program. Look at the diagrams and the the two lists to see how the cube is defined. More complex figures will take more information.

The center of the cube is (0,0,0). This can be changed by the user. The variables XX, YY, and ZZ define the center that the user chooses.

There are four procedures in the program. They are treated as keywords. To display the object the user can type in directly, or in a program, DISPLAY3D. ROTX, ROTY, and ROTZ rotate the object in radians held by the variable ROT. The program says that ROT is 20, its wrong. It should be in radians, like PI/10. Just to refresh your memory PI/2 is 90 degrees.

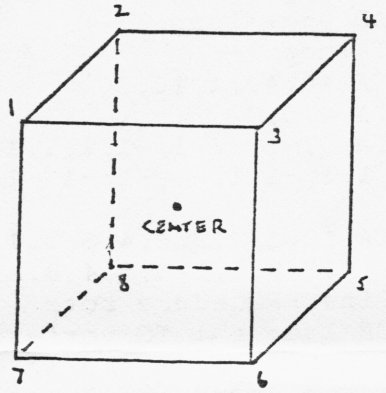
When a rotation procedure is called the object is spun in the axis defined. Since the Z axis is pointing out of the screen, rotating the object will make it appear to spin like it is on a turntable and you are looking down on it. I hope this is clear.

The variable D2 changes how large the object is on the screen. The larger the number the smaller the object.

When you run the program as it is, nothing appears to happen. All it has done is to load the array with the data and set up the variables. Type ROTX, ROTY, and then DISPLAY3D. You will now see the cube at an angle.

Have fun with the program. If you have any questions feel free to contact me. Its hard to summarize a lot of material in a short article.

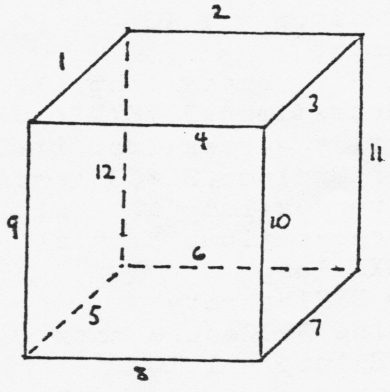
VERTICES



VERTEX LIST

1= (-1,-1,1)	2= (-1,1,1)
3= (1,-1,1)	4= (1,1,1)
5= (1,1,-1)	6= (1,-1,-1)
7= (-1,-1,-1)	8= (-1,1,-1)

EDGES

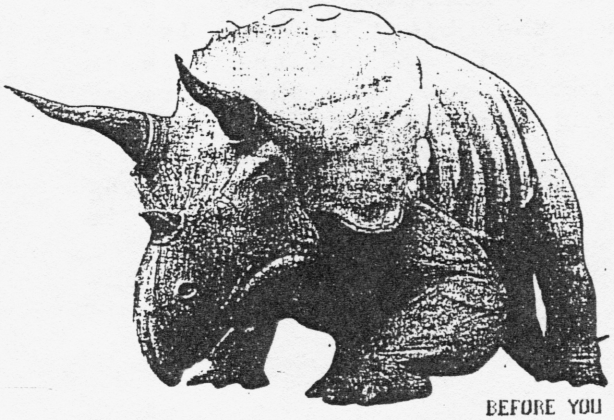


EDGE LIST

1= (1,2)	2= (2,4)
3= (4,3)	4= (3,1)
5= (7,8)	6= (8,5)
7= (5,6)	8= (6,7)
9= (7,1)	10= (6,3)
11= (5,4)	12= (8,2)

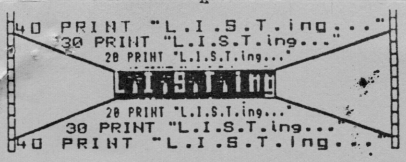
```
100 DIM vertex(100,3)
110 DIM edge(200,2)
190 RESTORE
200 READ xcenter, ycenter
210 READ d1, d2, rot
220 READ xx, yy, zz
230 READ vertexcount
240 FOR loop = 1 TO vertexcount
250  FOR loop2 = 1 TO 3
260    READ vertex(loop,loop2)
270  NEXT loop2
280 NEXT loop
290 READ edgecount
300 FOR loop = 1 TO edgecount
310  READ edge(loop,1)
320  READ edge(loop,2)
330 NEXT loop
1000 DATA 40,40,1,10,20
1010 DATA 0,0,0
1020 DATA 8,-1,-1,1,-1,1,1,1,-1,
1,1,1,1,1,1,-1,1,-1,-1,-1,-1,-1,
-1,1,-1
1030 DATA 12,1,2,2,4,4,3,3,1,7,8
,8,5,5,6,6,7,7,1,6,3,5,4,8,2
9000 DEFine PROCedure rotz
9010  FOR loop = 1 TO vertexcount
9020    x1 = vertex(loop,1)*COS
(rot)-vertex(loop,2)*SIN(rot)
9030    y1 = vertex(loop,1)*SIN
(rot)+vertex(loop,2)*COS(rot)
9040    vertex(loop,1) = x1
9050    vertex(loop,2) = y1
9060  NEXT loop
9090 END DEFine rotz
9100 DEFine PROCedure roty
9110  FOR loop = 1 TO
 vertexcount
9120    x1 = vertex(loop,1)*COS
(rot)-vertex(loop,3)*SIN(rot)
9130    z1 = vertex(loop,1)*SIN
(rot)+vertex(loop,3)*COS(rot)
9140    vertex(loop,1) = x1
9150    vertex(loop,3) = z1
9160  NEXT loop
9190 END DEFine roty
9200 DEFine PROCedure rotx
9210  FOR loop = 1 TO
 vertexcount
9220    y1 = vertex(loop,2)*COS
(rot)-vertex(loop,3)*SIN(rot)
9230    z1 = vertex(loop,2)*SIN
(rot)+vertex(loop,3)*COS(rot)
9240    vertex(loop,2) = y1
9250    vertex(loop,3) = z1
9260  NEXT loop
9290 END DEFine rotx
9300 DEFine FuNction transx
 (x,d1,d2)
9305  LOCal xprime
9310  xprime = (x*d2)/d1
9315  RETurn xprime
9320 END DEFine transx
9330 DEFine FuNction transy
 (y,d1,d2)
9335  LOCal yprime
9340  yprime = (y*d2)/d1
9345  RETurn yprime
9350 END DEFine transy
9400 DEFine PROCedure
 display3d
9405 CLS
9410  FOR loop = 1 TO edgecount
9420  LET point1 = edge(loop,1)
9430  LET point2 = edge(loop,2)
9435  LET x1 = vertex(point1,1)
 - xx
9440  LET x1 = transx(x1,d1,d2)
9445  LET x2 = vertex(point2,1)
 - xx
9450  LET x2 = transx(x2,d1,d2)
9455  LET y1 = vertex(point1,2)
 - yy
9460  LET y1 = transy(y1,d1,d2)
9465  LET y2 = vertex(point2,2)
 - yy
9470  LET y2 = transy(y2,d1,d2)
9480  LINE x1+xcenter,y1+ycenter
 TO x2+xcenter,y2+ycenter
9490  NEXT loop
9495 END DEFine display3d
```


DON'T BE A BONEHEAD



SUPPORT YOUR USER GROUP

BEFORE YOU BECOME EXTINCT



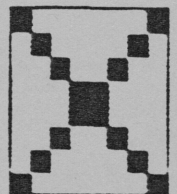
USER DEFINED GRAPHICS MADE EASY

BY CEDRIC R. BASTIAANS

If you're like me you have many more books on the TS2068 than just the User Manual that came with your computer. And maybe you have read everything on User Defined Graphics in these books. Maybe even given it a try but gave up because it was just too much trouble? All those binary numbers to be punched in! Error prone, what?

Well, weep no more, friends. The listing on the right is a program that was inspired by publications of the 2 authors shown in line 1. I combined the good things in their programs, added instructions and a few lines which make the new program run better and be of more help.
I submitted a simpler version to LIST, on tape, in the form of a contribution to a library tape. It was never included in any of the finalized library tapes and I have no idea what happened to my contribution! But here then, is the complete listing.

Key it in and RUN. You will be greeted by a brief explanation; follow the prompt. A square is drawn on screen and as you work with it you will find that it is an enlarged 8x8 matrix for the creation of a PIXEL MAP. Try the number keys 1 through 8 and notice how easy it is to lay down a black (ink) square or remove it again. Notice that when you have moved the arrow down, additional text appears which gives you instructions how to proceed when you have completed your pixel map. Ignore it for now and make the map shown in FIG. 1.



129
66
36
24
24
36
66
129

UDG HAS BEEN STORED AS GRAPHIC A
FIGURE 1

Then press ENTER and watch the computer list all the bit information necessary to record it in the computer, in decimal numbers and not in binary! When prompted for a character to carry this UDG, key in the character "A". You can choose any character between A - U (A and U included).
The computer will get back to you with the

```
  1 REM *** UDGMAKER ***
by CEDRIC R. BASTIAANS, inspired
by JOHN WENTWORTH (SUN VOL.2 #3)
and BILL POWERS (CATS NEWSLETTER
MAY 85)
 10 PRINT AT 2,1;"THIS PROGRAM
PRESENTS AN EASY";AT 3,1;"  WAY
TO MAKE  USER DEFINED";AT 4,8;"G
RAPHICS (UDG's)"
 12 PRINT AT 6,0;"THE UDG's WIL
L NOT  BE LOST WITH""NEW"", ""CL
EAR"", ""SAVE"" OR ""LOAD""ONLY
""RAND USR 0"",  RESETTING OR TU
RNING OFF THE COMPUTER WILL!"
 15 PRINT AT 11,3;"PRESS ANY KE
Y  TO CONTINUE": PAUSE 0: CLS
 20 PLOT 95,55: DRAW 65,0: DRAW
 0,65: DRAW -65,0: DRAW 0,-65
 30 POKE 23658,8: DIM A$(8,8)
LET R=1: FOR L=0 TO 7: POKE USR
"A"+L,255: NEXT L
 100 PRINT AT R+6,10;">"
 105 PRINT AT 0,1;"USE KEYS 1 TH
ROUGH 8 TO DARKEN THE APPROPRIAT
E SQUARE. AGAIN TOCANCEL YOUR CH
OICE."
 107 PRINT AT 4,0;"MOVE THE ARRO
W WITH KEYS A AND Z"
 110 PAUSE 0: LET I$=INKEY$
 120 IF I$>="1" AND I$<="8" THEN
 LET A$(R,VAL I$)=(" " AND A$(R,
VAL I$)="■")+("■" AND A$(R,VAL I
$)=" "): PRINT AT R+6,12;A$(R)
GO TO 110
 130 PRINT AT R+6,10;" ": LET R=
R+(1 AND I$="Z" AND R<8)-(1 AND
I$="A" AND R>1)
 135 PRINT AT 16,0;"IF YOU ARE S
ATISFIED WITH YOUR  DESIGN. PRES
S <ENTER> AND THE  COMPUTER WIL
L PRINT THE DECIMAL NUMBERS FOR
YOU TO RECORD FOR  FUTURE USE."
 140 IF CODE I$<>13 THEN GO TO 1
00
 200 DATA 128,64,32,16,8,4,2,1
 210 FOR J=1 TO 8: LET L=0: REST
ORE 200: FOR R=1 TO 8: READ X: L
ET L=L+(X AND A$(J,R)="■"): NEXT
 R: PRINT AT J+6,23;L: NEXT J
 220 PRINT AT 16,0;"
                                 "
 222 PRINT AT 16,0;"WHAT KEY DO
YOU WANT TO CARRY  THIS NEW CHA
RACTER? (A-U ONLY!)"
 225 INPUT B$
 230 FOR J=1 TO 8: LET L=0: REST
ORE 200: FOR R=1 TO 8: READ X: L
ET L=L+(X AND A$(J,R)="■"): NEXT
 R: POKE USR B$+J-1,L: NEXT J
 250 FLASH 1: BEEP .5,12: PRINT
AT 19,0;"UDG HAS BEEN STORED AS
GRAPHIC ";B$
 300 PRINT #1;AT 1,0;"MODIFY COP
Y ANOTHER QUIT?": PAUSE 0: FLASH
 0: IF INKEY$="M" THEN PRINT #1;
AT 1,0;"
                                 ":
 GO TO 100
 303 IF INKEY$="C" THEN COPY
 305 IF INKEY$="Q" THEN STOP
 310 FLASH 0: CLS: GO TO 10
 320 SAVE "UDGMAKER" LINE 1
```

indication that the UDG has been stored and you will be asked to either modify, copy, quit or do another UDG. I suggest you press Q and try your UDG in actual print by keying in SHIFT 9 (that's the GRAPHICS mode) and then "A" (or if you chose a different character, key in that character). If you keep the key down you'll see that this UDG could make a simple but nice border.

NOTE: If, after trial printing your UDG, you need to modify it, key in GO TO 100 to get the original pixel map back. After (C)opy, it will be erased and return you to the start of the program.

Now, if you need this UDG in your program, use the following program lines (we'll assume that you have chosen the character "A" for this UDG storage):

```
 1 DATA 129,66,36,24,24,36,66,129
 2 RESTORE 1:FOR N=0 TO 7: READ D: POKE USR "A"
+ N, D: NEXT N
```

Each time you use the graphics "A" in your program, the large X will be printed.

UDG's may be combined to make much larger graphics. Try the following:

Key in the pixel map shown in FIG 2; store "D"
Next, key in the map shown in FIG 3; store "O"
Then, key in the map shown in FIG 4; store "K"
Next, key in the map shown in FIG 5; store "S"
Next, key in the map shown in FIG 6; store "H"
Furthermore, the map shown in FIG 7; store "U"
Also, key in the map shown in FIG 8; store "N"
Finally, do the map shown in FIG 9; store "T"

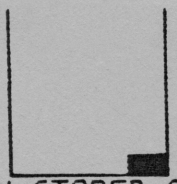
When you have finished this, CLS and enter the following in the direct mode:

```
PRINT AT 7,10;"D" (ENTER)
PRINT AT 7,11;"O" (ENTER)
PRINT AT 7,12;"K" (ENTER)
PRINT AT 7,13;"S" (ENTER)
PRINT AT 8,10;"H" (ENTER)
PRINT AT 8,11;"U" (ENTER)
PRINT AT 8,12;"N" (ENTER)
PRINT AT 8,13;"T" (ENTER)
```

NOTE: ALL THESE STRINGS ARE GRAPHIC CHARACTERS'

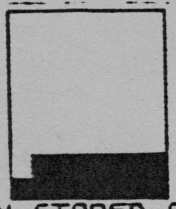
When you did everything right you will see on your screen the realistic figure of a dachshund' Like this:

Have fun with this program'



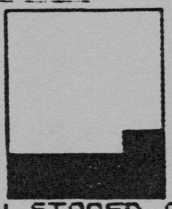
0.
0
0
0
0
0
3

UDG HAS BEEN STORED AS GRAPHIC D
FIGURE 2



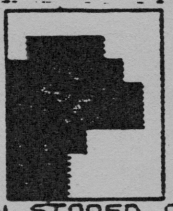
0
0
0
0
0
0
127
255

UDG HAS BEEN STORED AS GRAPHIC
FIGURE 3



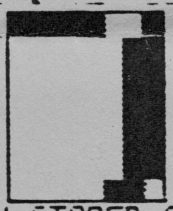
0
0
0
0
0
3
255
255

UDG HAS BEEN STORED AS GRAPHIC K
FIGURE 4



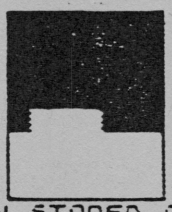
0
120
252
254
254
240
224
224

UDG HAS BEEN STORED AS GRAPHIC S
FIGURE 5



249
3
3
3
3
3
3
6

UDG HAS BEEN STORED AS GRAPHIC H
FIGURE 6

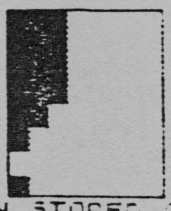


255
255
255
255
135
0
0
0

UDG HAS BEEN STORED AS GRAPHIC U
FIGURE 7

255
255
255
255
255
7
3
1

UDG HAS BEEN STORED AS GRAPHIC N
FIGURE 8



224
224
224
224
192
128
0
128

UDG HAS BEEN STORED AS GRAPHIC T
FIGURE 9

COMLINK I helps you spend more time on-line, less time waiting.

Now available for your ZX-81, T/S 1000, or T/S 1500, COMLINK I contains all the electronics and firmware needed to operate your Sinclair as an RS-232 compatible terminal.

You don't have to load the software from a cassette as it is always ready in COMLINK I's internal memory.

Features include: Expanded character set, Macro keys, Control key, Editing cursor, and Auto repeat.

COMLINK I works with any 300 baud modem or TNC. You can:

* Use COMLINK I with a TNC on packet radio.

* Access your company or school mainframe computer from home

* Access on-line bulletin boards and data bases such as The Source or CompuServe.

All operating power is supplied from the Sinclair. Simply plug COMLINK I into the expansion port of your Sinclair and connect it to an RS-232 compatible modem or TNC. One 'USR' call gets you started!

Each unit is attractively cased, fully tested and guaranteed, and is sent with complete operating instructions. Only $54.95 shipping included.

To order, send check or money order to:

A. Eckhardt
918 Anna Street
Boalsburg, PA. 16827

** PRESS RELEASE **

*** ZX-TERM*80 ***

No screen-dump could do justice to this program. This is THE definitive modem terminal program for the ZX81/TS1000 and TS1500.

* Westridge or Byte-Back modems supported.
* Instantly select 40 column, 60-column, or 80-column character widths. You read it right! 80 columns on the ZX81/ TS1000! Upper/lower case, and all ASCII symbols, while preserving Sinclair graphics; can even be customized to mimic IBM or other graphics sets.
* Two window modes. "One-window" mode displays all text on a single large screen; "Three-window" mode has separate windows for menus/prompts, your typed output, and received input! Windows 2 and 3 are even adjustable, just like on the "big guys."
* Allows XMODEM up/downloading of ASCII files, Sinclair programs, EVEN SINCLAIR VARIABLES! Usable with word-processors and similar programs.
* TS2040 and "big" printers supported, via any of the popular interfaces.
* Revolutionary modem input routine allows you to adjust windows, change width mode, etc. without losing a single incoming character.
* Relocatable, and usable with 16, 32, or 64K RAM.
* Takes only 4K at the top of memory, plus the 8K SCRAM board (or eq.)
* Review files in either Sinclair or ASCII mode.
* Allows local echo to be enabled or disabled.
* "Beginner-friendly" while maintaining full flexibility for customizing and "hacking."
* Redefined, auto-repeating Keyboard works like a true upper/lower-case terminal.

Finally, a REAL terminal package for your computer. Comes with a very comprehensive manual; it even has a large section on "Useful POKEs" for customizing to your tastes.

The price is the clincher...ONLY $24.95 !!

The Peak of Quality... from SILICON MOUNTAIN COMPUTERS

C-12, Mtn. Stn. Group Box, Nelson BC V1L 5P1, CANADA (604)352-1668

S.L.U.G.
4122 Wallingford Ln.
Louisville, Ky.
40218 2365
USA

LOUISVILLE
PM
NOV 10
1987
KY.

11-12-87

DON LAMBERT
3310 Clover Dr. S.W.
CEDAR RAPIDS, IOWA
52404